(2)

FAULT TOLERANT REAL-TIME SYSTEMS

Contract Number:
N00014-92-J-1771

YEARLY REPORT

1 October 1992 - 30 September 1993

Prepared for:

Chief of Naval Research
Code 4411/Annual Report
Ballston Tower One
800 North Quincy Street
Arlington, Virginia 22217-5660

Prepared By:

John P. Lehoczky, Principal Investigator
Department of Statistics
Carnegie Mellon University
Pittsburgh, PA 15213
(412) 268-8725

Principal Investigator Names:

J. Lehoczky	(412) 268-8725	jpl@k.cs.cmu.edu
L. Sha	(412) 268-5875	lrs@sei.cmu.edu
D. Siewiorek	(412) 268-2570	dps@a.cs.cmu.edu
J. Strosnider	(412) 268-6927	jks@usa.ece.cmu.edu
H. Tokuda	(412) 268-7672	hxt@k.cs.cmu.edu

PI Institution: Carnegie Mellon University
Contract Title: Fault Tolerant Real-Time Systems
Contract Number: N00014-92-J-1771
Reporting Period: 1 Oct 92 - 30 Sep 93

1 Productivity Measures

- Papers submitted but not yet accepted: 6
- Refereed papers accepted and in press: 17
- Refereed papers published: 15
- Books submitted or published: 0
- Book chapters or other articles: 3
- Ph.D. dissertations: 3
- Patents filed or granted: 0
- Invited presentations[1]: 15
- Contributed presentations: 5
- Honors, Prizes, Awards and Professional Activities:
  - **John Lehoczky**:
    - Associate Editor, Journal of *Real-Time Systems*,
    - Member of the program committee of the 14th IEEE Real-Time Systems Symposium, the 1993 ICDCS and the second Rate Monotonic Users Conference.
    - Member, NIH Special Study Section on Statistics (Chair, July, 1993)
  - **Lui Sha**
    - Member NASA Space Station Advisory Committee,
    - Chairman of the Board of Visitors of RICIS, an R&D center established by NASA and NASA JSC at University of Houston at Clearlake.
    - General chair, 13th IEEE Real-Time Systems Symposium,
    - Associate Editor, *Real-Time Systems*
    - Associate Editor, *IEEE Computer*
    - The paper: "Distributed System Design Using Generalized Rate Monotonic Theory," by L. Sha and S. Sathaye published in *Proceedings of The 2nd International Conference on Automation, Robotics, and Computer Vision*, 1992, was selected as one of the most innovative papers. An updated version will be

[1]The invited and contributed presentations exclude all conference and workshop proceedings which are listed under Publications.

published again in a Special Issue of the *Journal of Integrated Computer-Aid Engineering* in 1993.

- **Jay Strosnider**
  - Promoted to associate professor of electrical and computer engineering, July, 1993.
  - Member of the Program Committee, 13th IEEE Real-Time Systems Symposium
  - Program chair, Workshop on Real-Time Multimedia Systems, December, 1993.
- **Hide Tokuda**
  - Program Committee Member:
    - IEEE 11th IEEE Workshop on Real-Time Operating Systems and Software
    - International Symposium on Object Technologies for Advanced Software (ISOTAS '93)
    - WISS'93 (Workshop on Interactive Systems and Software), JSSST
    - 3rd International Workshop On Responsive Computer Systems
    - IEEE 12th IEEE Workshop on Real-Time Operating Systems and Software
    - 6th EUROMICRO Workshop on Real Time Systems

- Graduate students supported: 2
- Post-docs supported: 0
- Minorities supported: 0

Principal Investigator Names:

J. Lehoczky	(412) 268-8725	jpl@k.cs.cmu.edu
L. Sha	(412) 268-5875	lrs@sei.cmu.edu
D. Siewiorek	(412) 268-2570	dps@a.cs.cmu.edu
J. Strosnider	(412) 268-6927	jks@usa.ece.cmu.edu
H. Tokuda	(412) 268-7672	hxt@k.cs.cmu.edu

PI Institution: Carnegie Mellon University
Contract Title: Fault Tolerant Real-Time Systems
Contract Number: N00014-92-J-1771
Reporting Period: 1 Oct 92 - 30 Sep 93

2 Summary of Technical Progress

2.1 Overview

The ART (Advanced Real-Time Technology) Project of Carnegie Mellon University is engaged in wide ranging research on hard real-time systems. The project has as its overall goal the development and demonstration of predictable and fault tolerant hard real-time computer systems. To achieve this goal, research is being conducted in three interrelated areas:

1. The development of a theory of hard real-time resource management which includes processors, operating systems and communications which will permit the straightforward integration of predictable systems using open system standards.
2. The design and construction of operating systems that support the theory of hard real-time resource management.
3. The design of fault tolerance techniques including hardware and software fault tolerance using temporal redundancy and analytic redundancy to permit the construction of real-time systems whose performance and dependability are predictable.

The ART Project is supported, in part, by three distinct ONR Contracts (N00014-92-J 1771, N00014-92-J-1524 and N00014-91-J-1304). In this report, we describe progress for the principle investigators supported by these three contracts.

During the October 1, 1992 - September 30, 1993 period, substantial progress was made in each of these broad categories. Only the progress on real-time resource management and temporal redundancy for fault tolerance is briefly described below. A more detailed collection of briefing materials for the entire project is contained in the yearly *ART Project Review* provided to ONR representatives.

In July 1993, NGCR has asked us to 1) evaluate the real-time extention to IEEE Scalable Coherent Interface, which is an advanced computer plane that can support multiple topology using fiber optic connections. 2) to lead the technical effort for Navy's next generation high performance network. The theoretical work developed by ART Project researchers will serve as a foundation for these efforts.

2.2 Integrating Scheduling and Fault Tolerance

Over the last year, substantial progress was made on the integration of real-time scheduling with fault tolerance to create a theory of temporal redundancy. Temporal redundancy is an approach to real-time system dependability in which subtasks which are a part of tasks with real-time requirements but which fail their acceptance test can be executed again, the latter execution being scheduled so that, if successful,

the task will meet its timing requirements, and the reexecution does not cause any other task to miss its deadline. The failures occur randomly, thus they create, in effect, a stream of aperiodic job requests. The job requests correspond to the time required to retry the subtask, or an alternative version of the subtask, and they have a deadline which is the same as the deadline of the failed task. The temporal redundancy problem, therefore, can be considered to be a special version of the problem of jointly scheduling hard deadline periodic tasks and aperiodic tasks. However, in this case, the aperiodic tasks also have hard deadlines, a problem which has never been addressed in the rate monotonic environment. The goal of the research is to develop methods to solve this new joint scheduling problem and to assess the efficacy of the algorithms produced in enhancing real-time system fault tolerance. The joint scheduling problem with hard deadline aperiodic tasks was solved in the recent paper by Ramos-Thuel and Lehoczky to appear in the 1993 Real-Time Systems Symposium. The use of these methods to provide the largest possible temporal redundancy was studied in the Ph.D. dissertation of Ramos-Thuel.

Temporal redundancy in real-time systems requires that time be allocated to aperiodic tasks in such a way that they can meet their timing requirements without causing any non-failed task to miss its deadline. There are two static allocation algorithms for fixed-priority systems that have been proposed: the *Private Reservation Algorithm* (PRA) which reserves time which is bound to the recovery of individual tasks and the *Communal Reservation Algorithm* (CRA) which reserves a pool of time available to recovery operations on a first-come first-serve basis. The PRA permits certain tasks to have guaranteed recovery properties; however, the absence of resource sharing makes this algorithm inefficient in the sense that some tasks receive no additional coverage. The CRA provides resource sharing, but the pool of available time is created under worst case conditions. Consequently, while this ensures that no other tasks will miss their deadlines, the conservative calculations create situations when sufficient time is available for recovery, but the time provided by the CRA is inadequate.

The approach to improve upon the conservative CRA is to use the *slack stealing algorithm*. This algorithm makes detailed calculations of the slack that is available during any interval of time using the exact schedulability equations associated with fixed priority scheduling algorithms. When any aperiodic task is ready for execution, an exact calculation is made to see if there is sufficient time available to execute that task without missing any other deadlines. The performance of the *slack stealing algorithm* is different for hard deadline aperiodics than it is for soft deadline aperiodics. In the former case, there is no strongly optimal scheduling algorithm. Because the scheduling problem is on-line and the aperiodics have hard deadlines, a decision to accept one aperiodic task for processing may entail rejecting another task. A different algorithm may not be able to accommodate the first task but is, therefore, unable to accommodate the second. This makes the performance of these two algorithms incomparable. A second difference is that there is no optimal priority level at which to process the aperiodic tasks, whereas for soft deadline aperiodics, it is optimal to execute them at the highest priority level. This choice creates additional variability into the algorithmic structure. Nevertheless, the slack stealing algorithm is far superior to the PRA and CRA algorithms.

Unfortunately, in some cases the slack stealing method introduces a large memory and scheduling overhead. A direct extension of slack stealing for the hard aperiodic scheduling case yields a worst-case scheduling overhead of n^3, where n is the number of periodic tasks.

To reduce the implementation overhead of the slack stealing method, an algorithm called the *Myopic Slack Management* (MSM) algorithm was introduced. Although the MSM algorithm is also based on the concept of slack stealing, the memory overhead is reduced by using conservative estimates of the slack available for each periodic task at run-time. To make slack estimation computationally feasible, the

accumulation of available slack is restricted to relatively short intervals of time. As a result, it is said that the MSM algorithm is nearsighted, or myopic, in its ability to accumulate slack, but on the other hand, it is not as conservative as the CRA. The scheduling overhead is reduced by restricting the service of hard aperiodic tasks to a maximum of two priority levels; that of the failed periodic task issuing the recovery request, and the deadline monotonic priority level for the aperiodic. These techniques reduced the memory and scheduling overheads to a worst-case complexity of n.

The performance of the MSM algorithm may be lower than that expected from a direct implementation of the slack stealing approach because of its tendency to underestimate the slack available and the limitations imposed on the priority levels considered for service. However, the MSM trades off some of the performance of the slack stealing method for a scheduling solution which has significantly less overhead. A quantitative comparison of the performance of the static and dynamic allocation strategies was performed. Specifically, the comparison studies included the static Private and Communal Reservation Algorithms and the dynamic Myopic Slack Management algorithm.

To measure the effectiveness of an allocation algorithm, a metric referred to as *recovery coverage* was introduced. Recovery coverage parallels the well-known concept of error detection coverage. It was empirically computed as the percentage of recovery requests accepted for service relative to the total number of recovery requests issued during a simulation. Under no conditions was the service of a recovery request allowed to jeopardize the timing correctness of any fault-free application task. Analytical models for predicting the coverage provided by the PRA and the CRA were derived. The predicted coverage for the PRA was shown to match the empirical results very closely. The prediction model for the CRA was shown to be optimistic but it offered insights in explaining the performance behavior of this algorithm. All simulation results obtained for the application workloads considered were consistent. The MSM algorithm proved to be very robust to changes in periodic loading conditions and to increases in the size of the transient recovery load. The preallocation algorithms rarely came close to providing the high coverage observed for the MSM algorithm. Although the coverage estimates for the PRA remained stable as the size of the transient recovery load was increased, its coverage was highly sensitive to the periodic load. The CRA, on the other hand, was less sensitive to changes in the periodic load but its performance degraded significantly as the size of the recovery load was increased. In general, the performance of these preallocation algorithms was competitive with that of the MSM algorithm only in cases in which the joint processing load was small. Most of the reported coverage values represent steady-state performance estimates, that is, estimates of the coverage provided by an algorithm when the transient recovery load is observed to persist for an infinite period of time. Since transient recovery loads only exist for short periods of time, the sensitivity of coverage to finite transient durations was investigated. Results showed that although coverage tends to increase as the duration of the transient decreases, the rate of change is very small. Hence, steady-state coverage values can be considered adequate approximations to the coverage observed for finite-duration transients, albeit slightly conservative.

The research presented above was done in the context of temporal redundancy where the hard deadline aperiodic tasks arise when hard deadline periodic tasks fail their acceptance test. This creates a context in which the deadlines are relatively short. The methodology is applicable in situations with longer aperiodic deadlines. We are currently comparing the performance of fixed priority and dynamic priority slack stealing algorithms.

Principal Investigator Names:

J. Lehoczky	(412) 268-8725	jpl@k.cs.cmu.edu
L. Sha	(412) 268-5875	lrs@sei.cmu.edu
D. Siewiorek	(412) 268-2570	dps@a.cs.cmu.edu
J. Strosnider	(412) 268-6927	jks@usa.ece.cmu.edu
H. Tokuda	(412) 268-7672	hxt@k.cs.cmu.edu

PI Institution: Carnegie Mellon University
Contract Title: Fault Tolerant Real-Time Systems
Contract Number: N00014-92-J-1771
Reporting Period: 1 Oct 92 - 30 Sep 93

3 Transitions and DoD Interactions

ART project personnel frequently interact with DoD representatives, especially Lui Sha in his dual role as a member of the ART project and of the CMU SEI. Dr. Sha is deeply involved with transitioning rate monotonic scheduling theory to industry and government. His efforts include:

- Member, NASA Space Station Advisory Committee,
- Interaction with the Navy NGCR,
- Named Chairman of the Board of Visitors of RICIS, an R&D center established by NASA and NASA JSC at University of Houston at Clearlake
- Coordinated the real-time version of POSIX,
- Worked with IEEE 802.6 standards group to develop a real-time capability,

In addition, Hide Tokuda, as developer of ARTS (and Real-Time Mach), interacts regularly with NOSC, IBM and University of Virginia to coordinate the development of testbeds at all four sites and experimentation with ARTS.

As a part of our software fault tolerance effort supported by N00014-92-J-1524, we have interacted with MITRE Corporation to investigate the use of analytic redundancy for airborne radar tracking systems.

Finally, the rate monotonic scheduling theory is increasingly being adopted by major projects. These projects include:

- Navy BSY-1 and BSY-2,
- NASA Space Station Freedom (for system integration),
- European Space Station (recommended its use for its Hard Real-Time OS project).

Jay Strosnider interacts frequently with NRaD San Diego Distributed Combat Control project transitioning technology into Navy lab testbeds in San Diego. He also interacts with IBM, Bellcore and Intel on commercial applications of the developed technologies.

Principal Investigator Names:

J. Lehoczky	(412) 268-8725	jpl@k.cs.cmu.edu
L. Sha	(412) 268-5875	lrs@sei.cmu.edu
D. Siewiorek	(412) 268-2570	dps@a.cs.cmu.edu
J. Strosnider	(412) 268-6927	jks@usa.ece.cmu.edu
H. Tokuda	(412) 268-7672	hxt@k.cs.cmu.edu

PI Institution: Carnegie Mellon University
Contract Title: Fault Tolerant Real-Time Systems
Contract Number: N00014-92-J-1771
Reporting Period: 1 Oct 92 - 30 Sep 93

4 Software and Hardware Prototypes

A variety of hardware and software prototypes are being developed as a part of the project and have been extensively reported including the ARTS and RT-Mach operating systems. The newest hardware prototypes involve experimental testbeds to test analytic redundancy as a method of achieving software fault tolerance. These experiments are reported in the annual report for ONR Contract N00014-92-J-1524.

Principal Investigator Names:

J. Lehoczky	(412) 268-8725	jpl@k.cs.cmu.edu
L. Sha	(412) 268-5875	lrs@sei.cmu.edu
D. Siewiorek	(412) 268-2570	dps@a.cs.cmu.edu
J. Strosnider	(412) 268-6927	jks@usa.ece.cmu.edu
H. Tokuda	(412) 268-7672	hxt@k.cs.cmu.edu

PI Institution: Carnegie Mellon University
Contract Title: Fault Tolerant Real-Time Systems
Contract Number: N00014-92-J-1771
Reporting Period: 1 Oct 92 - 30 Sep 93

5 Publications and Presentations

5.1 Journal Publications

- Y. Ishikawa and H. Tokuda, "Distributed Real-Time Programming Language: RTC++ (in Japanese)", *Journal of Japan Society for Software Science and Technology*, Vol. 9, No. 2, 1992.
- Y. Tobe, S.T.-C. Chou, H. Tokuda, "Video Transmission using an ARTS Resource Reservation Scheme (in Japanese)", *JIPS, SIGArch*, Vol. 92, No. 17, March, 1992.
- Y. Ishikawa, H. Tokuda, and C. Mercer, "An Object-Oriented Real-Time Programming Language", *IEEE Computer*, Vol.25, No.10, 1992.
- L. Sha and S. Sathaye, "A Systematic Approach to Design Distributed Real-Time Systems," *IEEE Computer*, Sept 1993.
- D. Katcher, H. Arakawa and J.K. Strosnider, "Engineering and Analysis of Fixed Priority Schedulers", *IEEE Transactions on Software Engineering*, September, 1993.
- J.E. Sasinowski and J.K. Strosnider, "A Dynamic Programming Algorithm for Cache/Memory Partitioning for Real-Time Systems," to appear *IEEE Transactions on Computers*,1994
- S.S. Sathaye, L. Sha and J.K. Strosnider, "Analysis of Reservation Based Dual Link Networks for Real-Time Applications," to appear *IEEE Transactions on Computers*, 1994
- L. Sha, R. Rajkumar and S. Sathaye, "Generalized Rate-Monotonic Scheduling Theory: A Framework for Developing Real-Time Systems", to appear *Proceedings of the IEEE*, January, 1994.
- M. Klein, J. P. Lehoczky and R. Rajkumar, "Rate Monotonic Analysis for Real-Time Industrial Computing Applications", to appear *IEEE Computer*, January 1994.
- J. Strosnider, J. Lehoczky and L. Sha, "The deferrable server algorithm for enhanced aperiodic respoonsiveness in real-time environments," to appear *IEEE Transactions on Computers*, 1994.
- M. Gonzalez Harbour, M. Klein and J. Lehoczky, "Timing analysis for fixed priority scheduling of hard real-time systems," to appear *IEEE Transactions on Software Engineering*, 1994.

5.2 Conference Papers

- Lehoczky, J.P. and Ramos-Thuel, S., "An optimal algorithm for scheduling soft-aperiodic tasks in fixed-priority preemptive systems," *Proceedings of the 13th IEEE Real-Time Systems Symposium*, December 1992.
- S.S. Sathaye, L. Sha and J.K. Strosnider, "Scheduling Real-Time Communications on Dual Link Networks," *Proceedings of the 13th IEEE Real-Time Systems Symposium*, December 1992.
- C.W. Mercer, and H. Tokuda, "Preemptability in Real-Time Operating Systems", *Proceedings of 13th IEEE Real-Time Systems Symposium*, December, 1992.
- S.Savage and H.Tokuda, "RT-Mach Timers: Exporting Time to the User", to appear *Proceeding of USENIX 3rd Mach Symposium*, Apr. 1993.
- T.Nakajima, T.Kitayama and H.Tokuda, "Experiments with Real-Time Servers in Real-Time Mach", *Proceedings of USENIX 3rd Mach Symposium*, Apr. 1993.
- H. Arakawa, D.I. Katcher, J.K. Strosnider, H. Tokuda "Modeling and Validation of the Real-Time Mach Scheduler", *Proceedings of ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, May 1993.
- S.S. Sathaye, W.S. Kish and J.K. Strosnider, "Responsive Aperiodic Services in High Speed Networks," *Proceedings of the 13th International Conference on Distributed Computing*," May 1993.
- T. Kitayama, T. Nakajima, and H.Tokuda, "RT-IPC: An IPC Extension for Real-Time Mach", *Proceeding of USENIX 2nd Microkernel and Other Kernel Architecture Symposium*, Sept. 1993.
- F. Jahanian, R. Rajkumar and S. Fakhouri, "Processor Group Membership Protocols: Specification, Design and Implementation", to appear in the *Proceedings of the Symposium on Reliable Distributed Systems*, Oct. 1993.
- M. Bodson, J. Lehoczky, R. Rajkumar, L. Sha, M. Smith and J Stephan, "Software fault-tolerance for control of responsive sytems," to appear in the *IEEE Conference for Decision and Control*, December, 1993.
- S. Ramos-Thuel and J. Lehoczky,"On-line scheduling of hard deadline aperiodic tasks in fixed-priority preemptive systems," to appear in *Proceedings of the 14th IEEE Real-Time Systems Symposium*, December, 1993.
- T. Nakajima, T. Kitayama, H. Arakawa, and H. Tokuda "Integrated Management of Priority Inversion in Real-Time Mach", to appear *Proceedings of 14th IEEE Real-Time Systems Symposium*, December, 1993.

5.3 Workshop Papers

- L. Sha, J. Lehoczky, M. Bodson, P. Krupp and C. Nowacki, "Responsive airborne radar systems," *Proceedings of the Second International Workshop on Responsive Systems*, October, 1992.
- Stephen T.-C. Chou, Hideyuki Tokuda, "Real-Time Communication with Deadline-Based Scheduling" *Proceedings of IEEE 12th Workshop on Real-Time Software and Operating Systems*, May, 1993.
- M. Bodson, J. P. Lehoczky, R. Rajkumar, L. Sha, M. Smith and J. Stephan, "Software Fault-Tolerance for Control of Responsive Systems", to appear in *Third Interational Workshop on Responsive Systems*, October 1993.

- C. W. Mercer, S. Savage and H. Tokuda, "Processor Capacity Reserves: An Abstraction for Managing Processor Usage", to appear *Fourth Workshop on Workstation Operating Systems*, October, 1993.
- H. Tokuda and T. Kitayama, "Dynamic QOS Control based on Real-Time Threads", to appear *Proceeding of 4th International Workshop on Network and Operating System Support for Digital Audio and Video*, November 1993.
- S. Oikawa and H. Tokuda,"User-Level Real-Time Threads: An Approach towards High Performance Multimedia Threads," to appear in *Proceeding of 4th International Workshop on Network and Operating System Support for Digital Audio and Video*, November, 1993.

5.4 Book Chapters or Articles

- K. Ekanadham, S. Gregor, K. Hiraki, R. A. Iannucci and R. Rajkumar, "An Architecture for Generalized Synchronization and Efficient Context-Switching", Chapter in book "Multithreaded Architectures", R. A. Iannucci, Ed., Kluwer Academic Publishers, to be published in 1993.
- J. Lehoczky, "Real-time resource management," to appear in *Encyclopedia of Software Engineering*," (J. Marchiniak, ed.), John Wiley, 1994.
- L. Sha and R. Rajkumar, "Generalized Rate-Monotonic Scheduling Theory," to appear in *Encyclopedia of Software Engineering*," (J. Marchiniak, ed.), John Wiley, 1994.

5.5 Reports Submitted for Publication

- C. Mercer, R. Rajkumar and H Tokuda, "Applying Hard Real-Time Technology to Multimedia Systems", submitted to *IEEE Workshop on the Role of Real-Time Computing in Interactive/Multimedia Systems*, 1993.
- S.S. Sathaye and J.K. Strosnider, "Conventional & Early Token Release Schedluling Models for the IEEE 802.5 Token Ring," submitted to *International Journal on Real-Time Computing*.
- S.S. Sathaye, D. Katcher and J.K. Strosnider, "Fixed Priority Scheduling with Limited Priority Levels," submitted to *IEEE Transactions on Computers*.
- J.K. Strosnider and C.J. Paul, "A Structured View of Real-Time Problem Solving," submitted to *AI Magazine*.
- D. Katcher and J.K. Strosnider, "Dynamic versus Fixed Priority Scheduling: A Case Study," submitted to *IEEE Transactions on Software Engineering*.
- S. Ramos-Thuel and J.K. Strosnider, "Scheduling Fault Recovery Operations for Time-Critical Applications," submitted to *Fourth IFIP Conference on Dependable Computing for Critical Applications*.

5.6 Ph.D. Dissertations

- Sandra Ramos-Thuel, "Enhancing Fault Tolerance of Real-Time Systems through Time Redundance", Department of Electrical and Computer Engineering, Carnegie Mellon University, May, 1993.
- Shirish S. Sathaye, "Scheduling Real-Time Traffic in Packet-Switched Networks", Department of Electrical and Computer Engineering, Carnegie Mellon University, August, 1993.

• C.J. Paul, "A Structured Approach to Real-Time Problem Solving", Department of Electrical and Computer Engineering, Carnegie Mellon University, August, 1993.